+1.800.267.0730 | INFO@DKL.COM

In-memory table manager

tableBASE® is an IBM mainframe real-time high performance in-memory table manager. As a mainframe optimization solution, tableBASE is ideal for organizations that need to get every ounce of power from their mainframe systems to maximize performance and transaction throughput, while minimizing system resource usage at the application level. It allows for more powerful and efficient applications while lowering mainframe TCO and reducing operational spend. It can also help to increase the timeframe between upgrade capital expense.

In-memory table manager

As an off-the-shelf solution, it is perfect for those organizations that have a need to focus their development efforts more on revenue-generating business activity, and less on supporting in-house developed table management utilities.

Db2 accelerator

Overall Db2 database performance can be improved by optimization at the application level. Db2 applications with very high transaction processing rates can be optimized to use fewer resources (CPU and I/O) while improving application performance. Applied to several resource-intensive applications will dramatically improve overall Db2 performance.

VSAM and IMS database accelerator

Overall VSAM or IMS performance can be improved by optimization at the application level. Applications with very high transaction processing rates can be optimized to use fewer resources (CPU and I/O) while improving application performance. Applied to several resourceintensive applications will dramatically improve overall VSAM or IMS performance

Batch application accelerator

Batch applications with very high transaction throughput rates can be optimized to use fewer resources (CPU and I/O) while improving application performance. Batch run times can be sharply reduced.

OLTP accelerator

OLTP applications with very high transaction throughput rates can be optimized to use fewer resources (CPU and I/O) while improving application performance. The time needed to complete each transaction can be reduced significantly, dramatically improving the overall system performance.

Figure 1: Applications run faster using tableBASE

Query accelerator

Systems that run both long-run queries (like those used for analytics) and short-run queries (like those used in transaction processing) often need two solutions. tableBASE is the ideal solution to improve access time for short-run queries.

Agile market-reactive application solution

Business rules are processed ultra-fast when they are embedded within rules processing applications. But updating rules requires application recompiles. tableBASE applications can run almost as fast asembedded-rules applications, but with rules externalized they can be updated in hours or days rather than weeks or months.
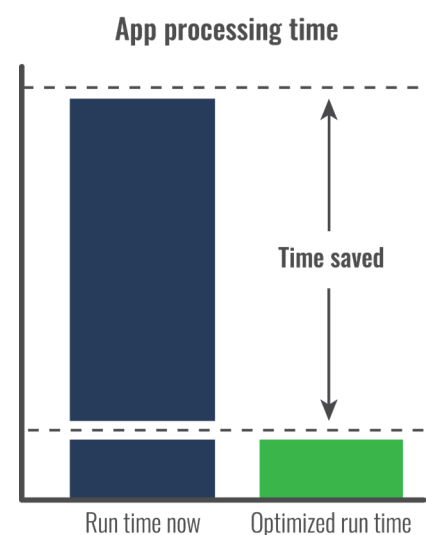
How tableBASE works

Generally, data access follows the 80 to 20 rule—80% of your data is accessed only 20% of the time, and 20% of your data is accessed 80% of the time. The 20% of your data that you are accessing 80% of the time is the reference data that you access many times for every business transaction.

This is typically where costs get eaten up, and represents a significant opportunity to increase performance and decrease operating costs — considerably. You can use DataKinetics tableBASE to do just this. Copying some of the most often accessed data into tableBASE highperformance in-memory tables allows you to access that data up to 100 times faster than is possible using any other method.

As shown in Figure 2, the data that is used most often during processing is accessed from tableBASE high-performance in-memory tables, while the rest of the data is accessed from DASD using current techniques. In this way, applications can run much faster—and the difference can be astounding.

Also, as part of transaction processing, temporary files are often created for the purpose of sorting: using temporary in-memory tables instead can save CPU resource usage and processing time for every business transaction.
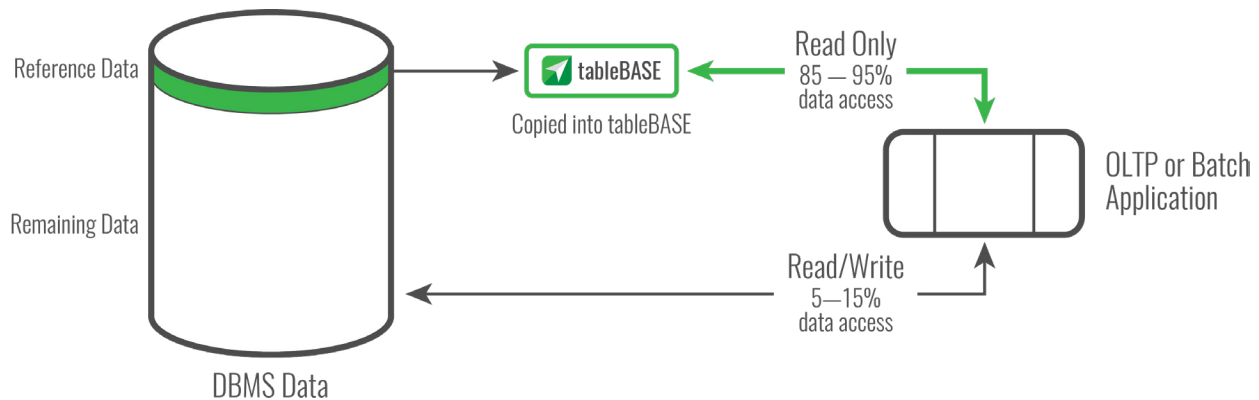

Figure 2: tableBASE augments your DBMS

tableBASE code path

tableBASE high-performance in-memory tables are accessed using the shortest possible code path—as close as possible to the calling application. Accessing buffered data uses the same code path that is used for accessing data from disk; it reduces I/O, but still uses the same code path. Figure 3 (top) shows an approximation of the Db2 code path taken for accessing data on disk and from buffers.
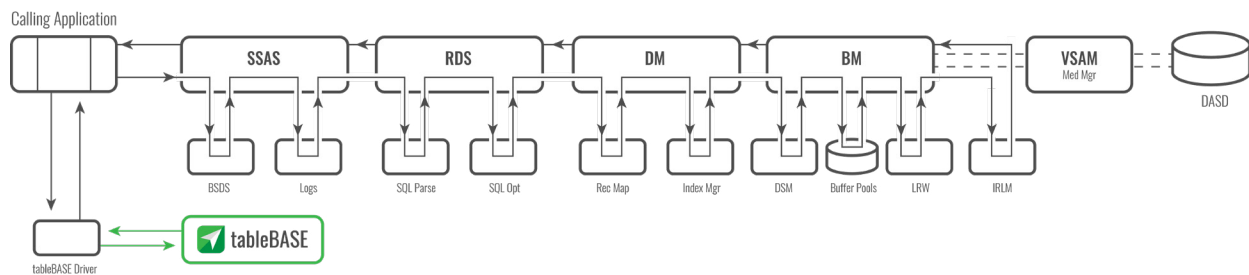

Figure 3: tableBASE code path

Figure 3 (bottom) shows the code path taken to access data from tableBASE high-performance in-memory tables. This is why accessing just a small amount of often-accessed data using tableBASE can make such a dramatic improvement to an application's processing speed.

Optional components

VTS

VTS augments the tableBASE core product by providing the capability to share table data. Using tableBASE, every region loads tables into a local table space region. After augmenting the core product with VTS, all regions can access the same data from a single shared region. Applications do not need to load data into their local table space, and system resource usage is optimized.

VTS Manager

VTS Manager provides the ability to schedule shared data updates and backups without shutting down transaction processing applications and without shutting down in-memory tables, providing continuous operation of DataKinetics optimization solutions.

VTS Gate

VTS Gate is an optional security enhancement solution for the tableBASE core product. With VTS Gate, only authorized users will be able to access in-memory data. Data integrity is also provided to protect in-memory data from malicious or accidental corruption.

CICS and IMS interfaces

Besides the default batch interface, there are two optional interfaces available. An IMS/TM interface provides compatibility for increased throughput IMS online applications, and a CICS interface provides compatibility for CPU-cycle and I/O-operation efficient CICS online applications.

Advanced interfaces for CICS and ISPF

Optional advanced tablesONLINE™ interfaces allow you to directly manipulate table data in memory, while in your production environment, eliminating many steps and providing much flexibility in both CICS and IMS environments.

tableBASE features

Accessing tableBASE

tableBASE is accessed via its API—a single call, and can be used in batch, CICS, IMS and other operating environments. This allows organizations to retain and optimize existing applications. A swap of I/O calls for tableBASE API calls will obtain up to a 100x improvement in application performance.

z/OS version compatibility

tableBASE operates on all version levels of the z/OS operating system.

Db2 version compatibility

tableBASE supports all version levels of Db2, and is designed to work in a Db2 stored procedure environment with multiple TCBs.

IMS version compatibility

tableBASE/IMS supports all version levels of IMS. ISPF version compatibility The optional tableBASE interface, tablesONLINE/ISPF supports all version levels of ISPF.

Sysplex support

tableBASE is fully capable of running in a Sysplex environment. Read-Only tables are fully supported across LPARs—updates to R-O tables are managed using Linear Data Set mappings. R/W tables can be managed using Db2 stored procedures. Datasets are fully supported across all MVS images in a GRS configuration.

CICSPLEX support

tableBASE is CICSPLEX compliant, but applications using tableBASE may require transaction affinities. This limitation can be eliminated by using the optional tableBASE VTS interface—this provides support for multiple regions accessing a shared Read/Write table. A transaction running in any region can create, update and access any table in a common shared region.

64-bit support

tableBASE is a 31-bit application; it does not use virtual storage above the 2G bar. z/OS fully supports 31-bit applications.
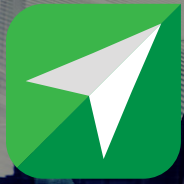
Other Features

tableBASE is a robust and mature in-memory table manager that has features that your DBMS does not have, including:

- In-memory table management that complements DBMS
- Date sensitive processing
- Indirect table access
- Flexible high-performance indexing
- Version control (up to 9 table generations)
- Multitasking – tableBASE is fully re-entrant
- In-memory repository for reference data
- Ideal holding area for temporary data
- Automatic table load and unload
- Multiple high-performance search methods
- Dynamic run-time table expansion

- Dynamic table reorganization
- Dynamic index creation and modification
- Dynamic run-time alternate views
- TSO/ISPF, CICS TS, IMS TM, Db2 SPAS compatibility
- C, C++, COBOL, PL/1, Assembler, Fortran accessibility
- Data access 1000x faster than I/O
- Data access 10x faster than DBMS buffer pooling
- Compatible with Db2, VSAM, BDAM, QSAM
- Compatible with MVS, IMS
- RACF compatibility